Hunter MT: A Course for Young Researchers in WMT17

Jia Xu and Yi Zong Kuang and Shondell Baijoo and Hyun Lee
and Uman Shahzad and Mir Ahmed and Meredith Lancaster and Chris Carlan
Hunter College, City University of New York
Jia.Xu@hunter.cuny.edu, {YiZong.Kuang38, shondell.baijoo41, hyun.lee16,
Uman.Shahzad34, Mir.Ahmed57, Meredith.Lancaster88, Chris.Carlan70}@myhunter.cuny.edu

Abstract

This paper documents an undergraduate course at Hunter College, in which one instructor, six undergraduates, and one high school student built 17 machine translation systems in six months from scratch. The team successfully participated in the second Conference on Machine Translation (WMT17) evaluation on the news task in Finnish-English and Latvian-English and on the bio-medical task in French-English, English-French, English-German, English-Romanian, and English-Polish.

1 Introduction

Machine learning has advanced the state-of-the-art of artificial intelligence at a rapid speed. There has been an increasing amount of related courses introduced. However, hands-on experience is of vital importance to novices (Lopez et al., 2013).

Through conventional education it may take many years for beginners to find a research direction in the field of machine translation. Introductory courses in this field can be either too theoretical or too detailed, leaving students lost in coding. Therefore, our goals were to make the material both detailed and comprehensive and also bring novelty and excitement into this course. We propose teaching methods that are centered around the machine translation competition. With this idea in mind we were able to achieve our goals because of three key factors. First, we were able to focus on the pragmatical aspects of the teaching material; second, the study was comprehensive, since we covered all the components of a machine translation system; and third, student motivation was enhanced, because the results were directly available in the MT community through the WMT evaluation. These key factors helped us to attain our goals optimally and efficiently.

We will discuss the teaching methods to introduce undergraduates to topics of advanced machine translation technology. We will describe the outcome, the machine translation systems in 17 languages, in particular the ones that were successfully submitted in the WMT17, as well as the research revised in this course. In conclusion, we will evaluate our course with feedback of students.

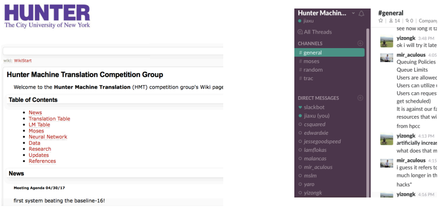
2 Backgrounds

At Hunter college, we experimentally designed an applied machine translation course to provide an opportunity for novice students to learn by competing against the best senior teams around the world in one of the most significant machine translation competitions, the WMT17. The students' levels ranged from high school to college senior, and none of them had any prior knowledge of machine learning or machine translation.

Hunter college offered supportive facilities for the course. Two students took an introductory C++ programming class (Software Analysis and Design I) (Hunter, 2016). Some students took an one-week "Linux introductory class" (Hunter, 2016) at the beginning of the semester. The open machine translation resources served as the basis of this course. "statmt.org" (SMT, 2017) provides excellent readings and software for the beginning student. "SMT Tutorial" (Knight, 1999) by Kevin Knight et. al. is an essential reading on machine translation, and the book "Statistical Machine Translation" (Koehn, 2010) by Philipp Koehn was recommended because it gives a more in-depth explanation.

We view machine translation as a high-dimensional, multiclass classification task. The reference book used was "Pattern Classification"

Proceedings of the Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 422–427
Copenhagen, Denmark, September 711, 2017. ©2017 Association for Computational Linguistics



HUNTER
The City University of New York

wiki: WikiStart

Hunter Machine Translation Competition Group

Welcome to the **Hunter Machine Translation** (HMT) competition group's Wiki page

Table of Contents

- News
- Translation Table
- LM Table
- Moses
- Neural Network
- Data
- Research
- Updates
- References

News

Meeting Agenda 04/30/17

first system beating the baseline-16!

see now long it ta

yizongk 3:48 PM
ok i will try it late

mir_aculous 4:05
Queuing Policies
Queue Limits
Users are allowed
Users can utilize
Users can reques
get scheduled)
It is against our fa
resources that wi
from hpcc

yizongk 4:13 PM
artificially increas
what does that m

mir_aculous 4:15
i guess it refers to
much longer in th
hacks*

yizongk 4:16 PM

(a) Hunter MT Wiki page (b) Chat room in Slack

Figure 1: Infrastructure for team coordination.

by Duda and Hart (Duda and Hart, 1973). With the insight in previous competitions, the instructor introduced basic methods on the blackboard in an interactive way, covering the following topics: Bayes Decision Rule, Maximum Likelihood, Word Alignment Models and Training, Search, Language Modeling, Cross-Validation, Domain Adaptation, Design Bagging, Neural Network, and Neural Machine Translation.

Team meetings took place weekly to discuss problems and solutions. Students set up a Wiki homepage called "Hunter MT" to share their work and post questions internally. A Slack Platform provided a coherent working atmosphere for students connecting with each other in real time. In our Slack board students had posted 13.9K messages and uploaded 131 files during this six-month course. Figure 1 contains screenshots of these homepages of the team.

The software development and machine translation systems were hosted in the High Performance Computing Center (HPCC) at the College of Staten Island (CSI) CUNY, located in New York City. Each student had their own account to conduct their experiments in their respective language pairs. Students shared their experiences within the team to avoid repeating experiments. Jobs were submitted and scheduled via queuing system. At HPCC, we used an infiniband cluster named Penzias and fat node server named Arrow. Penzias, which is a cluster, uses the Sandy Bridge chip and NVIDIA K20m GPU.

3 Core MT engine

We built phrase-based machine translation systems using the open software toolkit Moses (Koehn et al., 2007). We used an EMS script to run the translation pipeline, which includes preprocessing, word alignment training, tuning, testing, and error logs for debugging.

3.1 Pre- and Post-processing

For all language directions, we used the Moses default tokenization and true-casing tool. The preprocessing involved tokenization, truecasing, and cleaning. The experimental results showed that using truecasing produces a better result than not using it for most language directions.

3.2 Word alignment

Word alignments were generated based on GIZA++ (Och and Ney, 2000) and mGIZA (Gao and Vogel, 2008) for all language directions with the grow-diag-final option. We ran five iterations of Model 1 (Brown et al., 1993), five iterations of HMM (Vogel et al., 1996), and four iterations of IBM model-4 (Brown et al., 1993). Training sets included in-domain training data and selected out-of-domain training data that we will outline in detail for individual language pairs in Section 4 and Section 5.

We put a limit of 100 words maximum on the sentence length. For bio-medical tasks, the maximum sentence length was set to 80. The main reason for this was to shorten the time of the training process. Because there is more training data to handle in the bio-medical tasks than in the news tasks, considering both in-domain and out-of-domain corpora, we decided to place a heuristic threshold to shrink the training time to an acceptable one (a couple of days, depending on languages and processors in the HPCC).

3.3 Language model

The language models used were 7-gram SRILM (Stolcke, 2002) with Kneser-Ney smoothing (Kneser and Ney, 1991) and linear interpolation. Having the highest possible n-gram is generally good pratice, but due to limited time and the exponential rate of time needed to train the language model, we decided to use 7-gram to train the language model.

The English language model is shared among all foreign-language-to-English translation systems. It is a mixture language model with domain adaptation following (Xu et al., 2007). The language models trained on individual corpora is linearly interpolated on its n-gram probabilities. Their weights are optimized with respect to the perplexity on the development set of German-English newstest2016. The individual training corpora are Europarl v7, a fraction of common crawl (due to limited computational resource), news commentary 2007-2016, DCEP, LETA, FAREWELL, RAPID, and news discussion. In addition to those, we also used NYT and XIN from GigaWord (WMT, 2017). Other language models were trained with all available corpora same as in English.

3.4 Tuning

The tuning set is the development set of WMT16 for most language pairs. We used MIRA (Hasler et al., 2011) to tune single systems to find out the optimal feature weights in the log-linear combination. We used 100-best in the tuning, and we also heuristically tuned the Moses parameters, such as maximum phrase length, stack size in search, nbest size, maximum sentence length, and search pruning options. The remaining parameters followed the default values in the EMS (Koehn et al., 2007). To rank the system, we used the BLEU (Papineni et al., 2002) score.

Below we describe our machine translation systems for language directions we submitted in the WMT17 evaluation.

4 Data Sets and Settings for News Task

The translation systems for different language pairs are built with the same methods as in Section 3. However, they are trained on different parallel training set. Table 1 shows the corpus used in the GIZA++ (Och and Ney, 2000) training for machine translation systems for each language direction. For each WMT17 evaluation task we participated in, we computed the number of sentences, the number of running words for the training set, the development set and the test set, respectively as shown in Table 2. We also computed the OOV rate of the running word and the OOV rate of the vocabulary (Voc.) for the source and target test set in each system.

4.1 Finnish-English

The parallel training set included corpora of Europarl version 7, Rapid 2016, and Titles. The translation result was evaluated on the WMT17 news test set of 2016.

4.2 Latvian-English

The parallel training set included corpora of Europarl version 7, Rapid 2016, LETA, FAREWELL, and DCEP. The translation result was evaluated on the WMT News Test set 2016.

5 Data Sets and Settings for Bio-medical Task

Below I will describe our submission systems in the bio-medical task (Yepes et al., 2017).

5.1 English-German

We used the WMT17 provided corpora for training and tuning, including Europarl v7, News Commentary v12, Rapid Corpus of EU press releases, and parts of the Common Crawl corpus. We added the previous years' test sets in the training.

Ultimately, we found that for German, increasing the maximum sentence length and phrase length increased the BLEU score by a few points. We also found that setting the language model order to 7 helped the BLEU score by one point. While these optimized parameters slowed down the training of the system as expected.

5.2 English-Polish

Both in-domain and selected out-of-domain corpora were used. The list of in-domain corpora used in this experiment came from the following sources: CESTA, ECDC, EMEA (open subtitle and news crawl), Medical Web Crawl, Medical Web Text from CzEng 1.6, MuchMore, PatTR Medical, and Subtitles. For out-of-domain corpora, the sources were the following: Cordis, EU-bookshop, EUROPARL, JRC-Acquis, MultiUN, News Commentary, OpenSubtitles, PatTR, and Rapid. The combined corpora totaled 39,442,076 lines, with a total of 302 million words. To preprocess the corpora, we used default Moses tokenizing tools. The resulting cleaned corpora totaled 39,321,672 lines.

5.3 English-Romanian

Because the Romanian language uses the alphabet system, for the Romanian system, we used a setting similar to that used for the Polish system. The corpora consisted of in-domain sources, such as ECDC, EMEA, and Subtitles. It also included out-of-domain sources, such as: EURO-BookShop, EUROPARL, JRC-Acquis, and Open Subtitles. The resulting corpora totaled 62 million lines and 416 million words. After preprocessing, to deal with the unique symbols in the language and to conform to a standard format of the text,

ID	Languages	Domain	BLEU[%]	Corpora	Test set
1	English-German	News	26.28	Europarl,Global,NC,Rapid	News Test 2016
2	German-English	News	33.61		News Test 2016
3	English-Czech	News	13.59	Europarl,CommonCrawl,News'12	News Test 2016
4	Czech-English	News	15.48		News Test 2016
5	English-Russian	News	15.88	CommonCrawl,NC,Wiki	News Test 2016
6	Russian-English	News	26.23		News Test 2016
7	Turkish-English	News	12.48	SETIMES2	News Test 2016
8	English-Turkish	News	10.93		News Test 2016
9	Finnish-English	News	18.53	Europarl,Rapid,Titles	News Test 2016
10	English-Finnish	News	12.82		News Test 2016
11	Latvian-English	News	24.61	Europarl,Rapid,LETA,FAREWELL,DCEP	News Dev 2017
12	English-Latvian	News	18.43		News Dev 2017
13	English-French	Bio	25.16	Europarl,Medline,NC,Scielo	Health Test 2016
14	French-English	Bio	24.46		Health Test 2016
15	English-German	Bio	29.56	Europarl,NC,UFAL,ECDC, Subtitles,EMEA,PatTR,Medical	Himl Test
16	English-Polish	Bio	18.70	Europarl,ECDC,EMEA,EUBS Subtitles,Cordis,JRC,Rapid	Himl Test
17	English-Romanian	Bio	17.36	Europarl,ECDC,EMEA, Subtitles,EUbookshop,JRC	Himl Test

Table 1: Translation systems in different language pairs in BLEU-c [%].

such as true-casing, the corpora was reduced to 61 million lines, which is a significant reduction compared to English-Polish. Using SRILM, we built a 5-gram language model.

6 System Outputs

Table 1 shows our system outputs for different language directions in the news and in the biomedical domain. All translation systems were only generated in this course. Each student was responsible for the translation systems of the language direction that interested them. The results are produced based on the training and test corpora listed in the last two columns, respectively.

Machine translation systems are built by students with the guidance and assistance of the course instructor, Jia Xu. Each student worked on different language directions: Yi Zong Kuang (15,16,17), Shondell Baijoo (1,2,11,12), Hyun Lee (13,14), Uman Shahzad (6,7,9,10), Mir Ahmed (3,4), Meredith Lancaster (5,6), and Chris Carlan (11,12). Yi Zong Kuang and Shondell Baijoo contributed to the Human evaluation in the News Track. Yi Zong Kuang, Shondell Baijoo, Hyun Lee worked on system descriptions together with the course instructor. Mixture language models and some translation systems are conducted by the instructor as example experiments.

7 Research Components

We applied two methods to improve over baseline systems. These are course exercises without being included in the final submission.

7.1 Design bagging

We applied the bagging (Breiman, 1996) and its improved version design bagging (Papakonstantinou et al., 2014) to train the systems. As shown in Algorithm 1 and Algorithm 2, the parallel training set is sampled into $m = 30$ blocks (subsets or bootstraps), each block contains b parallel sentences which is 50% of the whole parallel training data. $x \in R[0, N-1]$ means to uniform randomly assign an integer value to x in the range from 0 to $N-1$, where N is the size of the training data. Either bagging, see Algorithm 1 or design bagging, see Algorithm 2 is used to construct the blocks. Then each of the 30 blocks was used to train a machine translation system, with the same setting as described in Section 3. We translated the test set with each of these systems and then combined all 30 translation results with a system combination tool (Heafield and Lavie, 2010) whose weights were tuned on the development set.

7.2 Phrase-based language model

We also applied a phrase-based language model. The likelihood of a sentence is based on decomposed phrases instead of single words, given histories. This is achieved by treating phrase segmentation as a hidden variable and developing a complete phrase-based n-gram LM that was tailored for machine translation use. The details of this algorithm are described in (Xu and Chen, 2015).

Algorithm 1 Bagging

```
Input: block size $b$, number of blocks $m$, number of elements $N$.
Initialize $m$ empty blocks.
for $k = 0$ to $m - 1$ do
    for $i = 0$ to $N - 1$ do
        $a[i] = i$
    end for
    for $i = 0$ to $2Nlog_2N - 1$ do
        $x \in R[0, N - 1]$
        $y \in R[0, N - 1]$
        Swap $a[x]$ and $a[y]$
    end for
    for $i = 0$ to $b - 1$ do
        $b[k][i] = a[i]$
    end for
end for
Output: $m$ blocks each with $b$ distinct elements.
```

Algorithm 2 Design Bagging

```
Input: block size $b$, number of blocks $m$, number of elements $N$.
Initialize $m$ empty blocks.
for $i = 1$ to $b \times m$ do
    select current smallest block (if not unique, choose randomly)
    $S_1 \longleftarrow$ the set of elements not in this block
    $S_2 \longleftarrow$ set of elements that among the elements in $S_1$ appears the minimum of times in other blocks
    Choose randomly an element from $S_2$ and put it into the current block
end for
Output: $m$ blocks each with $b$ distinct elements.
```

	Training Set		Dev Set		Test Set		OOV	
Languages	Sentences	Words	Sentences	Words	Sentences	Words	Words	Voc.
Latvian	4507745	56447016	2003	41245	974	21417	12.2%	5.8%
English	4507745	67601629	2003	49206	974	25496	8.0%	2.6%
Finnish	2633183	45235670	4500	72692	3002	46572	19.9 %	8.7 %
English	2633183	62847985	4500	98000	3000	64813	8.9 %	2.3 %
English	2794276	67279904	1000	21932	5023	140505	6.2 %	0.7 %
French	2794276	75320850	1000	27383	5023	192732	6.9 %	0.6 %
English	2061633	55855699	2495	45762	1931	34833	14.0 %	4.9 %
German	2061633	53356277	2495	43150	1931	35283	19.4 %	6.7 %
English	39321672	381409086	3922	69626	1931	34833	3.9 %	0.9%
Polish	39321672	307458011	7844	137396	1931	33527	3.5 %	1.3 %
English	61943814	536905597	3922	69626	1931	34833	4.0 %	0.9 %
Romanian	61943814	508776149	7844	137400	1931	37939	5.1 %	1.4 %

Table 2: Corpus statistics for various language directions

8 Teaching Outcome

As an outcome of the training, we performed an anonymous questionnaire on SurveyMonkey (surveymonkey, 2017) to evaluate the course and receive feedback. The overall rating of this course is satisfactory, with some comments for example: “The hands-on experience was by far the best.” and “Being able to see the work was very important.” In response to the question: What is the most valuable thing you learned? Students said ”Understanding how research is done.” At the same time, we also received such suggestions as “The tutorials and mini lectures were helpful and should be more frequent.” and “more on Neural Network Machine Translation”.

9 Summary

We described the teaching experience of a supervised study course of six undergraduates and a high school student. One course instructor guided young and fresh machine translation learners.

The teaching feedback is encouraging, and the products generated during this course were a cool surprise: 17 machine translation baseline systems and a successful participation of the WMT17.

Acknowledgments

We appreciate the strong support of the Department of Computer Science of Hunter College, the Office of the Dean of Arts and Science, as well as the Provost’s Office at Hunter College. In particular, we thank Professor William Sakas for his arrangement and effort to make this course happen and to help outreaching course participants. We also thank Professor Susan Eppstein for her insightful advice on the course management. We are thankful to the High Performance Computing Center at the City University of New York for providing rich computational resources and IT services. Students got hardware and software support from CUNY HPC staff as well as basic training about usage of HPC systems. The Computer Science Department of CUNY Graduate Center offered location to host our seminars and supported the continuation of our team. Above all, we would like to thank Lampros Flokas, Pablo Gonzalez, Liam Geron, and Hussein Ghaly for the insight they brought from the reading group, as well as the effort they put into the CUNY machine translation systems built by the graduate study group.

References

Leo Breiman. 1996. Bagging predictors. *Machine Learning* 24(2):123–140.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics* 19(2):263–311.

Richhard O. Duda and Peter E. Hart. 1973. *Pattern Classification and Scene Analysis*. John Wiley, New York, NY.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*. Association for Computational Linguistics, pages 49–57.

Eva Hasler, Barry Haddow, and Philipp Koehn. 2011. Margin infused relaxed algorithm for moses. In *The Prague Bulletin of Mathematical Linguistics*. pages 69–78.

Kenneth Heafield and Alon Lavie. 2010. Voting on n-grams for machine translation system combination. AMTA.

Hunter. 2016. The website of CSci 135 course software analysis and design I at Hunter college. http://catalog.hunter.cuny.edu.

Hunter. 2016. The website of Hunter beginners' linux class. http://www.hunter.cuny.edu/csci/pressroom/news/beginners-linux-class.

Reinhard Kneser and Hermann Ney. 1991. Forming word classes by statistical clustering for statistical language modelling. In *1. Quantitative Linguistics Conf.*. Trier, Germany, pages 221–226.

Kevin Knight. 1999. A statistical MT tutorial workbook. Http://www.isi.edu/natural-language/mt/wkbk.rtf.

Philipp Koehn. 2010. *Statistical machine translation*. Cambridge University Press.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL on Interactive Poster and Demonstration Sessions*. Association for Computational Linguistics, pages 177–180.

Adam Lopez, Matt Post, Chris Callison-Burch, Jonathan Weese, Juri Ganitkevitch, Narges Ahmidi, Olivia Buzek, Leah Hanson, Beenish Jamil, Matthias Lee, et al. 2013. Learning to translate with products of novices: a suite of open-ended challenge problems for teaching mt. *Transactions of the Association for Computational Linguistics* 1:165–178.

Franz Josef Och and Hermann Ney. 2000. GIZA++: Training of statistical translation models.

Periklis A Papakonstantinou, Jia Xu, and Zhu Cao. 2014. Bagging by design (on the suboptimality of bagging). In *AAAI*. pages 2041–2047.

Kishore A. Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*. Philadelphia, pages 311–318.

SMT. 2017. The homepage of statistical machine translation. Http://www.statmt.org.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of IWSLT*. Denver, Colorado, pages 901–904.

surveymonkey. 2017. Free online survey software & questionnaire tool. Https://www.surveymonkey.com.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *COLING96*. Copenhagen, Denmark, pages 836–841.

WMT. 2017. Conference on machine translation. Http://www.statmt.org/wmt17/.

Jia Xu and Geliang Chen. 2015. Phrase based language model for statistical machine translation. *arXiv preprint arXiv:1501.04324* .

Jia Xu, Yonggang Deng, Yuqing Gao, and Hermann Ney. 2007. Domain dependent statistical machine translation. In *In Proceedings of the MT Summit XI*. pages 515–520.

Antonio Jimeno Yepes, Aurélie Névéol, Mariana Neves, Karin Verspoor, Ondřej Bojar, Arthur Boyer, Cristian Grozea, Barry Haddow, Madeleine Kittner, Yvonne Lichtblau, Pavel Pecina, Roland Roller, Amy Siu, Philippe Thomas, and Saskia Trescher. 2017. Findings of the WMT 2017 Biomedical Translation Shared Task. In *Proceedings of the Second Conference on Machine Translation (WMT17) at EMNLP*. Copenhagen, Denmark.